

ECUSimLite GNU/Linux

User Manual



DEEP THOUGHT
S Y S T E M S P V T . L T D

Table of contents

1. Introduction	3
1.1. Modes	3
1.2. Kit Contents	3
2. ECUSimLite Block Diagram	4
2.1. Connection Sequence	4
2.2. Install Instructions	5
2.2.1. Supported Platforms	5
2.2.2. Install Instructions	5
2.3. Getting Started	7
2.3.1. Set Range	8
2.3.2. Set DTC	9
2.3.3. Clear DTC	10
2.3.4. Enable Logging	10
2.3.5. Testing OBD Software	11
3. Abbreviation	13

1. Introduction

ECUSimLite is a free PC application that can be used to test devices like OBD port readers, without the need to connect to a vehicle. It uses Deep Thought's CANMate device and CANMate API to implement ISO15765 compliant OBD II ECU simulation. Using the ECUSimLite along with the CANMate device can save 100s of dollars compared to stand alone hardware simulators which serve just the same purpose!

Features of the lite version:

- Supports mode1,3 and 4
- Message Logging

It requires special cabling and 12V Power supply. It can be used for testing OBD II devices and software.

1.1 Modes

Table 1.1 Modes

Mode(hex)	Description	What we support?
01	Show current data	PIDs supported are Speed, RPM, MAF, Fuel level and ECT
03	Show stored DTC	Setting DTC (currently one DTC supported)
04	Reset/Clear DTC	Clear DTC

1.2 Kit Contents

- CANMate
- ECUSimLite
- Custom Cable (either the customer can make this using the instructions downloadable from www.dthoughts.com or he/she can buy it from Deep Thoughts.)

12V Power Supply with appropriate current rating has to be procured by the customer.

2. ECUSimLite System Block Diagram

The ECUSimLite System includes 12V Power supply, CANMate, PC/laptop with ECUSimLite installed, Bluetooth OBD scan dongle, Android device, and custom cable. The connection sequence is described in the next section.

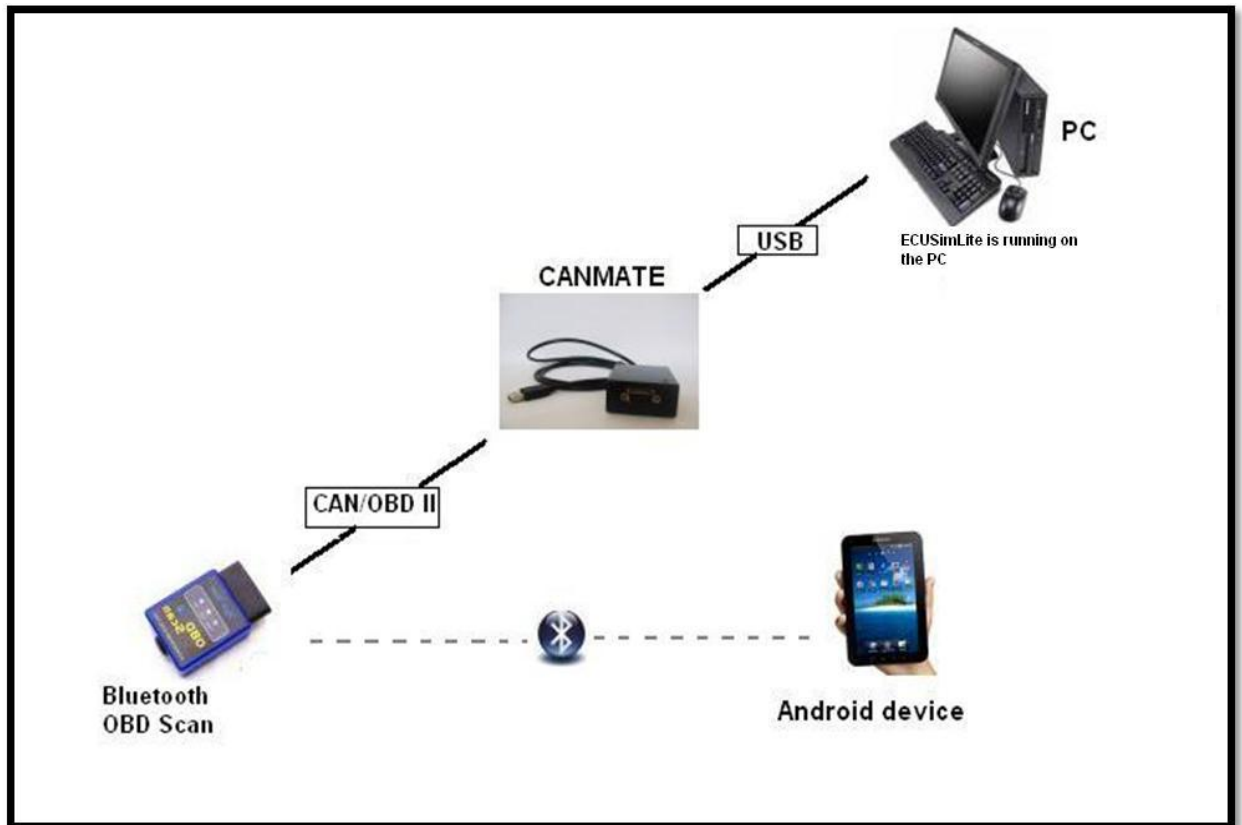


Figure 2.1 ECUSimLite System Block Diagram

2.1 Connection Sequence

- Connect CANMate device to USB port of PC/laptop via USB cable.
- The red LED turns on to indicate that the device is powered up.
- Connect custom cable to the CANMate device.
- Connect 12V Power supply to the custom cable.
- Connect Bluetooth OBD scan dongle to the other end of custom cable.
- Make sure that the power LED of the OBD scan is turned on.
- After connecting all devices, open the ECUSimLite application.

Note: The 12V Power supply is used to power up the Bluetooth OBD scan dongle.

2.2 Install Instructions

Install instruction ECUSimLite in GNU/Linux

2.2.1 Supported Platforms

Operating System : Ubuntu 13.04

Kernel Version : 3.8

Dependency : GTK3

2.2.2 Installtion Steps

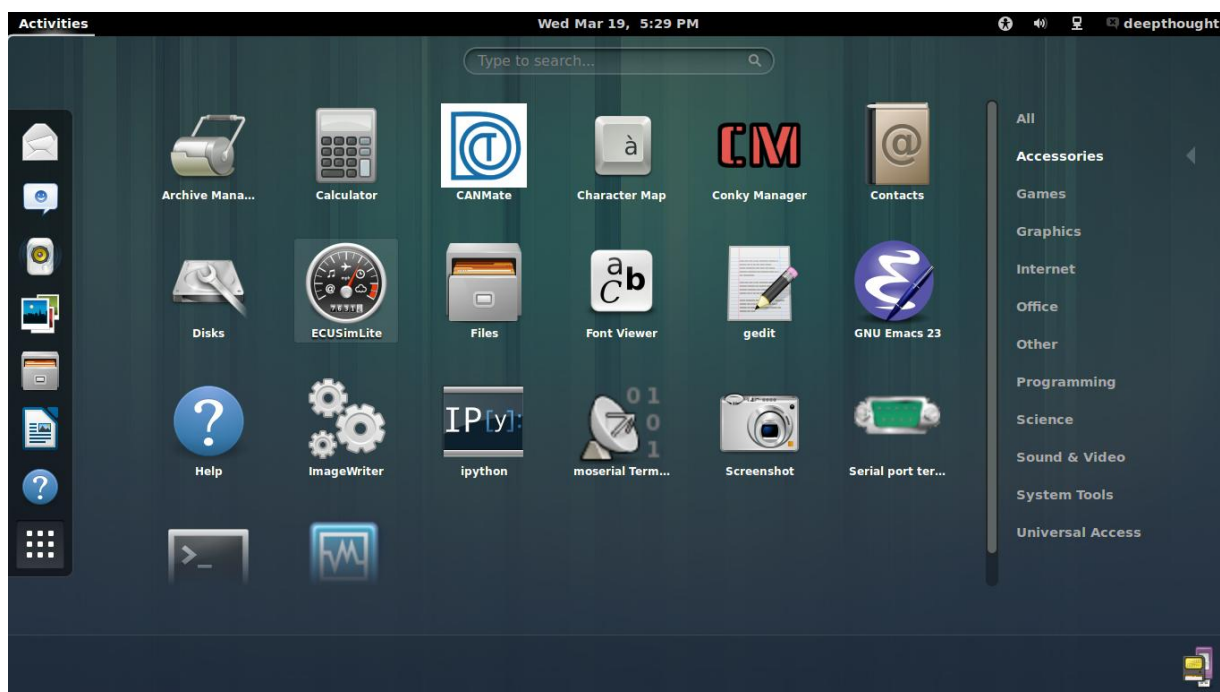
CANMate Linux must be preinstalled for running ECUSimLite application.

Download ECUSimLitev1.0.deb

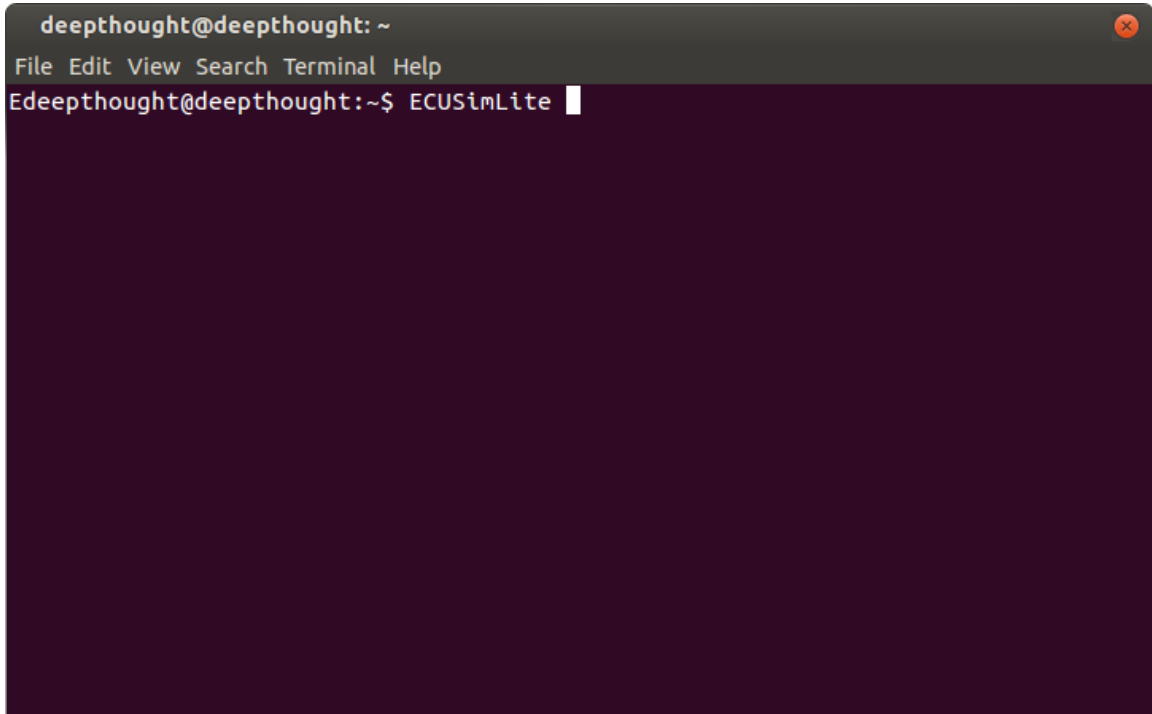
Install the deb file

```
sudo dpkg -i ECUSimLitev1.0.deb
```

The ECUSimLite App is listed in Accessories menu or you can run it from terminal by issuing the command "ECUSimLite "



Starting ECUSimLite from Terminal



```
deepthought@deepthought: ~  
File Edit View Search Terminal Help  
Edeepthought@deepthought:~$ ECUSimLite
```

2.3 Getting started

- Now, open the ECUSimLite application.
- The application will open the CANMate device and the green LED will turn on in the device to indicate that the port has been opened for communication.

The screenshot of the application is shown below.

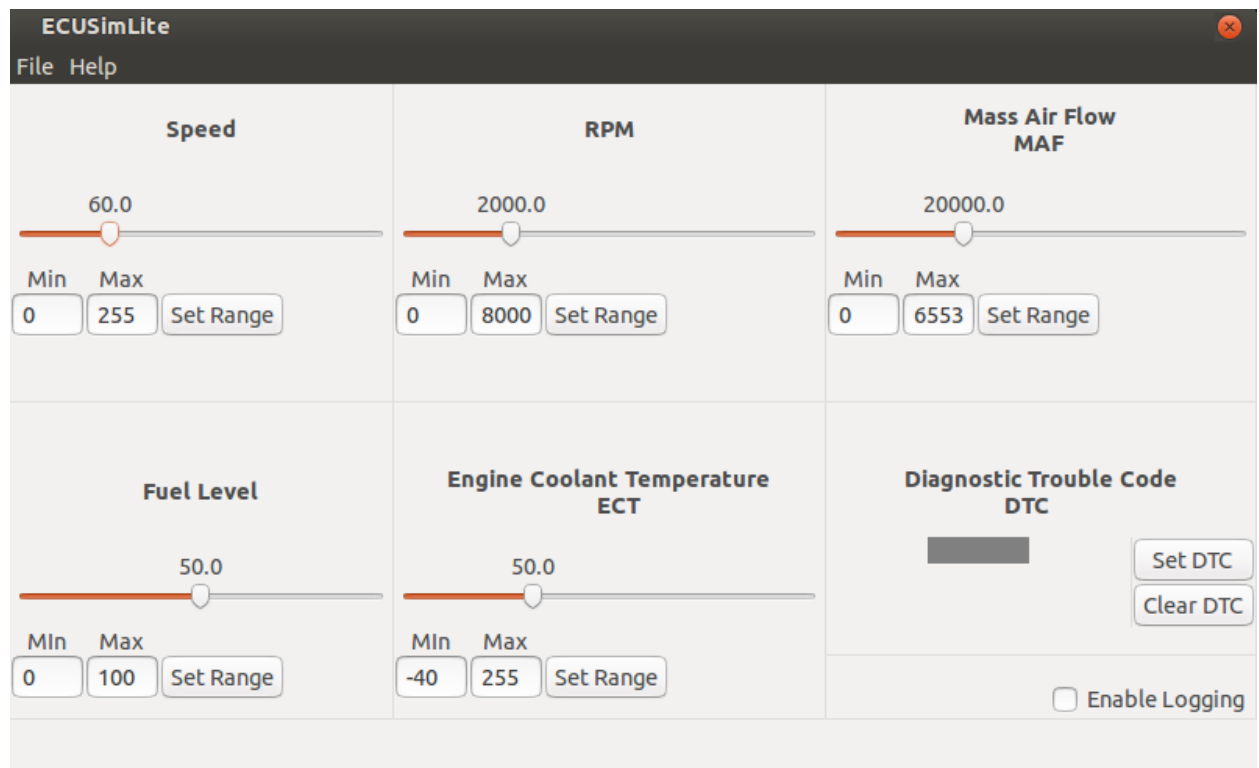


Figure 2.2 ECUSimLite App

ECUSimLite has the following capabilities:

- Set the range of the parameters using *Set Range* button.
- Change the value of the parameters using the *Knob*.
- Set the Diagnostic Trouble Code using *Set DTC* button.
- Clear the Diagnostic Trouble Code using *Clear DTC* button.
- The *Enable Logging* will save the communication details to ECUSim.csv file in hex format. The csv file will be saved in the Desktop.

2.3.1 Set Range Set Range

Each parameter has certain range as shown below:

Table 2.1.Parameter Table

Parameter	Min	Max
Speed	0	255
RPM	0	8000
MAF	0	65535
Fuel Level	0	100
ECT	-40	255

The *Set Range* is used to set the range of these parameters. It allows us to set the min and max values of parameters. The actual minimum and maximum values of the parameters are shown in the table above. We can specify the values for each parameter only within this limit.

Example: Setting the range of speed parameter

The minimum and maximum value of speed is provided in the [Table1.Parameter Table](#). Setting the min value to '0' and max value to '200':

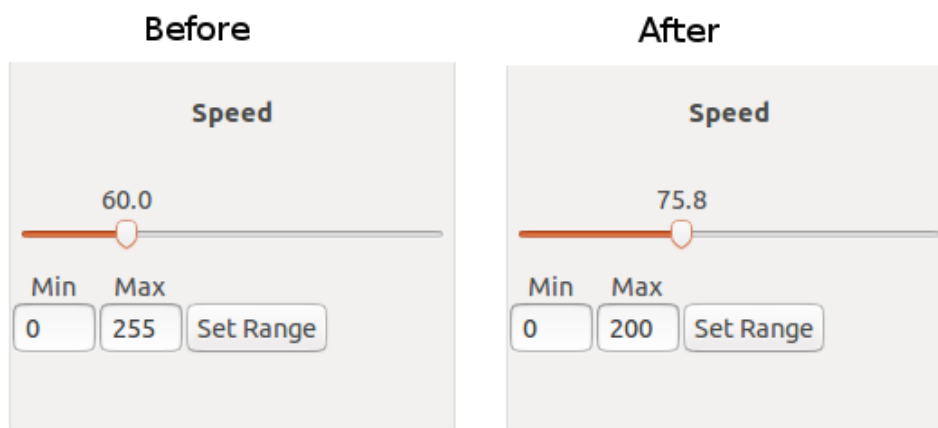


Figure 2.3 setting the speed parameter

Note: Never set the min value below 'actual min value' and max value above 'actual max value'. If we select the invalid range a warning dialogue box will appear as shown below.

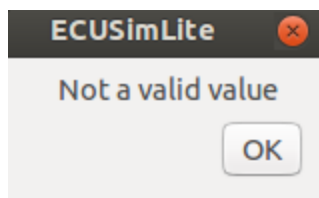


Figure 2.4 Invalid range warning

2.3.2 [Set DTC](#) [Set DTC](#)

The function of Set DTC is to set the diagnostic trouble code. Currently, we support only one DTC (P0143- Sensor Circuit Low Voltage (Bank I Sensor 3)).

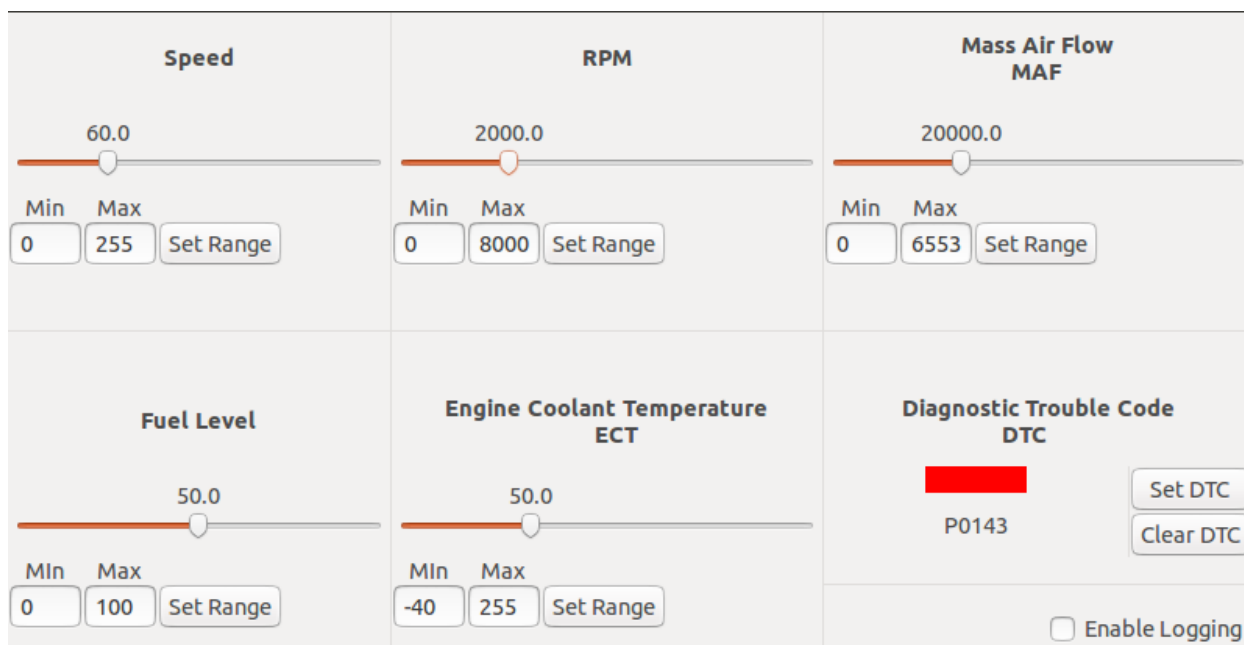


Figure 2.5 Set DTC

2.3.3 Clear DTC Clear DTC

It clears the diagnostic trouble code as shown in the figure.

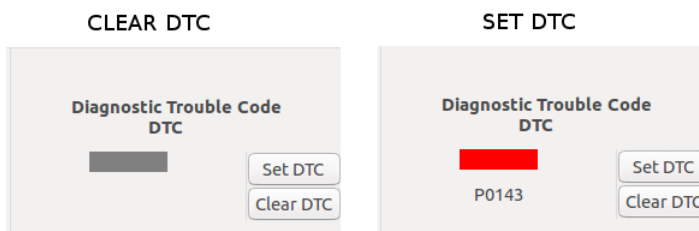


Figure 2.6 Set DTC and Clear DTC

2.3.4 Enable logging

By default, the ECUSimLite App will not log the incoming and outgoing messages. Logging can be enabled by checking the box. All the incoming and outgoing messages will be logged into *ECUSim.csv* file [The default location is Desktop“/home/USERNAME/Desktop”].

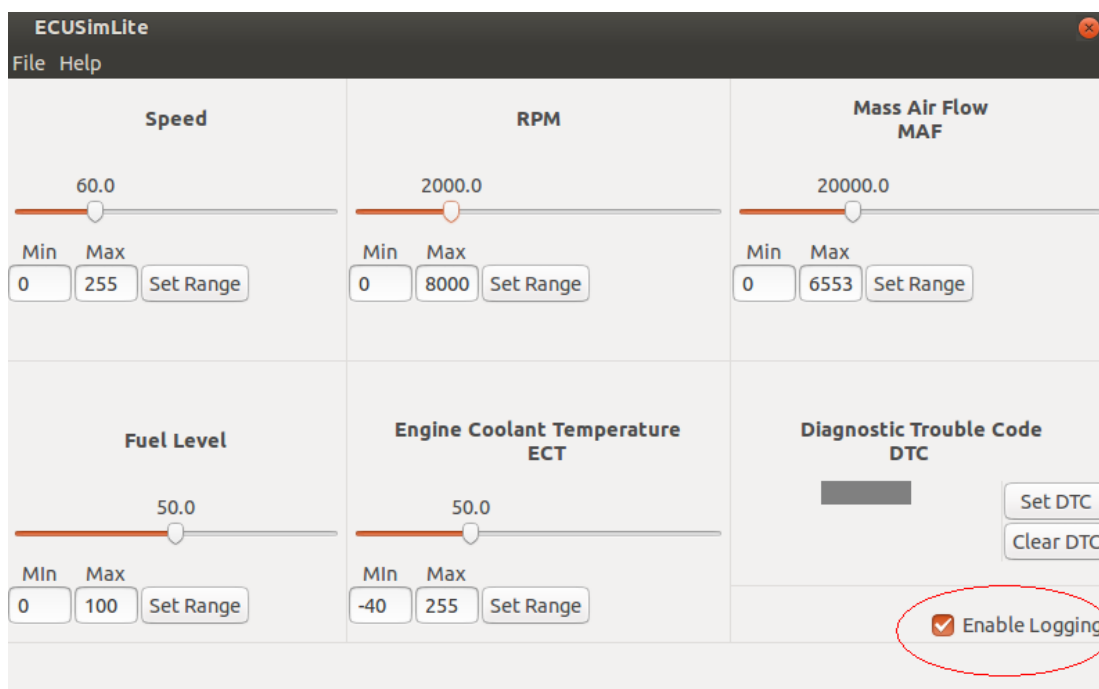


Figure 2.7 Enable logging

3. Testing OBD Software

An example is explained here to demonstrate the working of ECUSimLite System. The System includes a PC, OBD Scan dongle, CANMate device, Android device, and Power supply.

First of all, we have to establish all required connection as per the connection sequence in [section 2.1](#). Then open the ECUSimLite app and set values of parameters. After finishing all these steps, we can pair the OBD scan and android device (Here we used an android tablet).

After pairing up, open OBD software to test the working. Here, we use OBD Check software - "Torque". The *Torque* displays the same parameter values that we have set on ECUSimLite App. Also the OBD application displays the definition of DTC code.

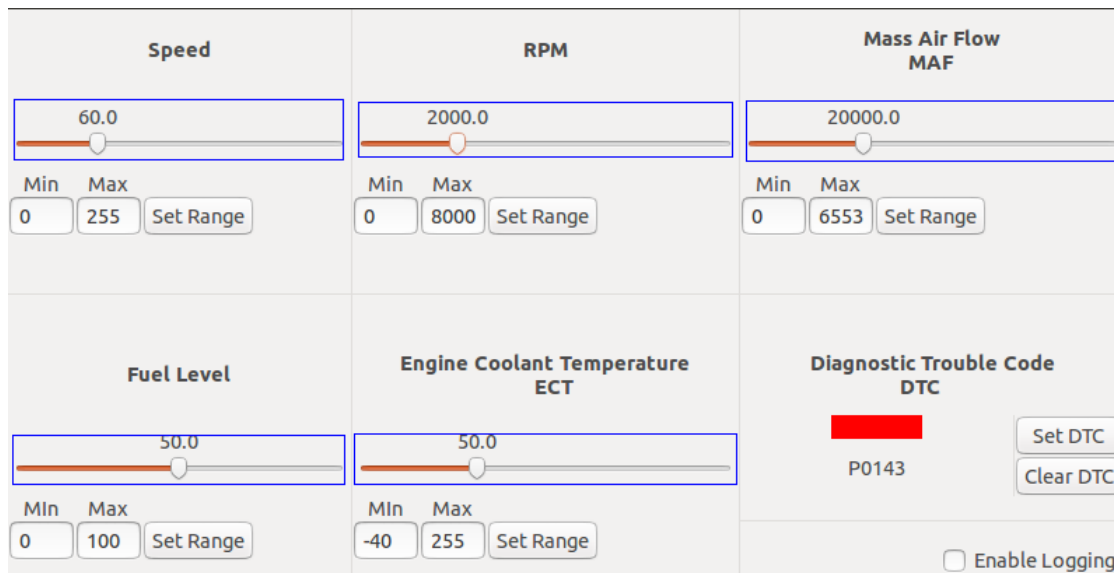


Figure 3.1 Set the PIDs

Here, as an example we set speed=60, RPM= 2000, MAF=20000.00, Fuel Level= 50, ECT= 50 in ECUSimLite App. It is shown in the [figure3.1](#). [Figure 3.2](#) shows set DTC.

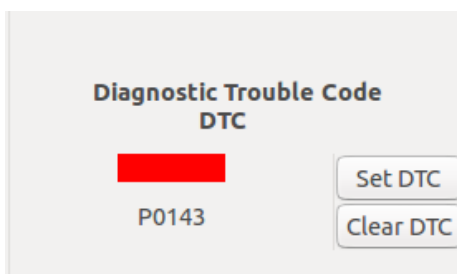


Figure 3.2

Then checked the *Torque*, It displays parameters as shown in the [figure3.3](#).



Figure 3.3 Torque result

To check the DTC, we used another OBD Software-“*OBD Auto Doctor*”. After setting the DTC ([Figure3.2](#)) in ECUSimLite App, we got the DTC definition in this OBD App as shown in the [figure3.4](#).

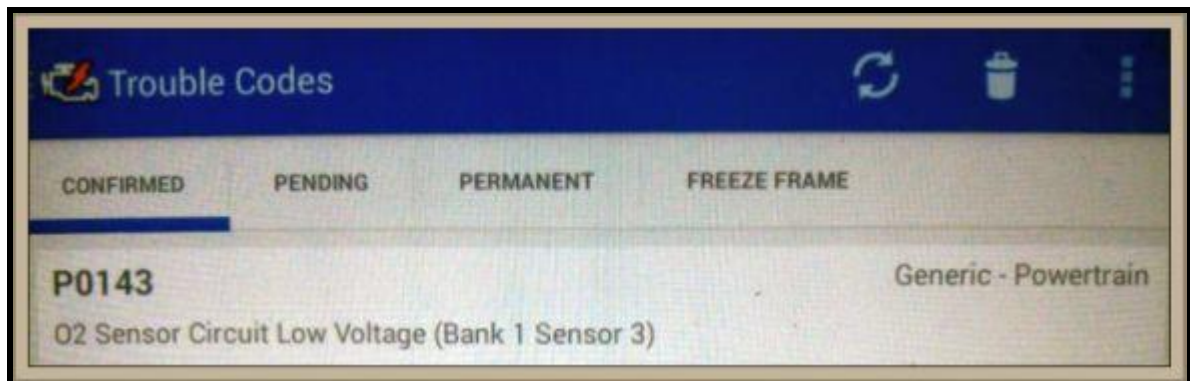


Figure 3.4

Abbreviations

CAN	Controller Area Network
DTC	Diagnostic Trouble Code
ECT	Engine Coolant Temperature
ECU	Engine Control Unit
MAF	Mass Air Flow
OBD	On-Board Diagnostics
PID	Parameter ID
USB	Universal Serial Bus

This user manual is published by:

Deep Thought Systems (P) Ltd,

"Suvarna", T.C.9/1857(2),
K-130, Kochar Road,
Sasthamangalam, Thiruvananthapuram,
Kerala, India. PIN-695010
Ph: +91-471-4066468
Mob: 9496253676
Email: info@dtthoughts.com

For more information on our products please visit www.dtthoughts.com

© Copyright 2013 Deep Thought Systems. All rights reserved. You are hereby granted a license to download and/or print a copy of this document.

Any rights not expressly granted herein are reserved.